FINAL REPORT

for the

HLLV AVIONICS REQUIREMENTS STUDY AND ELECTRONIC FILING SYSTEM DATABASE DEVELOPMENT

Contract No. NAS8 - 39215
(DR-3)

Prepared for:

National Aeronautics and Space Administration
Marshall Space Flight Center

Prepared by:

Smith Advanced Technology, Inc.

June 30, 1994

CONTENTS

FINAL REPORT

1.0 OBJECTIVE

This final report is intended to provide a summary of achievements and activities of Smith Advanced Technology, Inc. (SAT) in the performance of the effort described in the Statement of Work for Contract NAS8-39215, "HLLV Avionics Requirements Study and Electronic Filing System Database Development". The contract's objective was to explore a new way of delivering, storing, accessing, and archiving study products and information and to define top-level system requirements for Heavy Lift Launch Vehicle (HLLV) avionics that incorporate Vehicle Health Management (VHM). This report includes technical objectives, methods, assumptions, recommendations, sample data, and issues as specified by DPD No. 772, DR-3.

2.0 SUMMARY OF ACHIEVEMENTS AND ACTIVITIES

The following report is organized into two major subsections, one specific to each of the two tasks defined in the Statement of Work: the Index Database Task and the HLLV Avionics Requirements Task. The Index Database Task resulted in the selection and modification of a commercial database software tool to contain the data developed during the HLLV Avionics Requirements Task. Within each task's section, all summary information for the task is addressed.

2.1 INDEX DATABASE TASK

The index database task consisted of the effort to develop an electronic filing system (EFS) index database to contain information relating to NASA/MSFC presentation material and demonstrating use of the database by entering and accessing data for HLLV Avionics system requirements. SAT's significant activities and accomplishments for this task were as follows:

- Analyzed database functions and user interfaces using the Propulsion System Database (PSDB) prototype, as referenced in the statement of work.

- Identified and refined relational database entities, relationships, and attributes for a database based on the PSDB prototype.

- Developed a relational database model encompassing the database elements identified. Designed this model to serve as an ideal model with which to compare and select an integrated COTS database/Graphical User Interface (GUI) software tool or to use as a design for implementation of a custom database application with an associated custom GUI.

- Documented the relational database model in a design document, "Database Model Design for the HLLV Avionics Requirements Study and Electronic Filing System Database Development". Delivered draft 1 of this document to MSFC on June 3, 1993. Delivered the final version of this document (dated July 28, 1993) concurrently with the progress report (DPD No. 772, DR-2).

- After development of the model, performed an analysis of available image databases as integrated database/GUI tools. Determined that a COTS software tool would sufficiently satisfy the requirements and could be tailored to include the necessary database attributes from the model. Documented the results of this analysis in the database model design document.

- Using an evaluation version of the selected tool, "CompassPoint", by Northpoint Software, (with full functionality, but limited in capacity), developed a database tailored to the model as specified in the design document. Utilized SAT's in-house resources for this effort.

- Populated the tailored database with HLLV avionics requirements data from the avionics study task. Then demonstrated the database system to MSFC personnel. In addition, installed a copy of the demonstrated database on an MSFC Macintosh computer.

- During the presentation of database demonstrations a need for windows PC compatibility on a server-based network system was specified. Discussions with the vendor indicated that development of an improved version of the tool was in progress. This version would be windows-based with networking capability and would incorporate several enhancements as recommended by NASA/MSFC and SAT. Jointly decided with NASA/MSFC to select this tool as the basis upon which the Electronics Filing System Database would be implemented.

- During the development, SAT maintained visibility in the tool development process with the vendor to ensure requirements focus.

- After the vendor, Northpoint Software, completed the tool development, SAT acquired, tested, and demonstrated the new version, then applied the tailoring specified in the database model design.

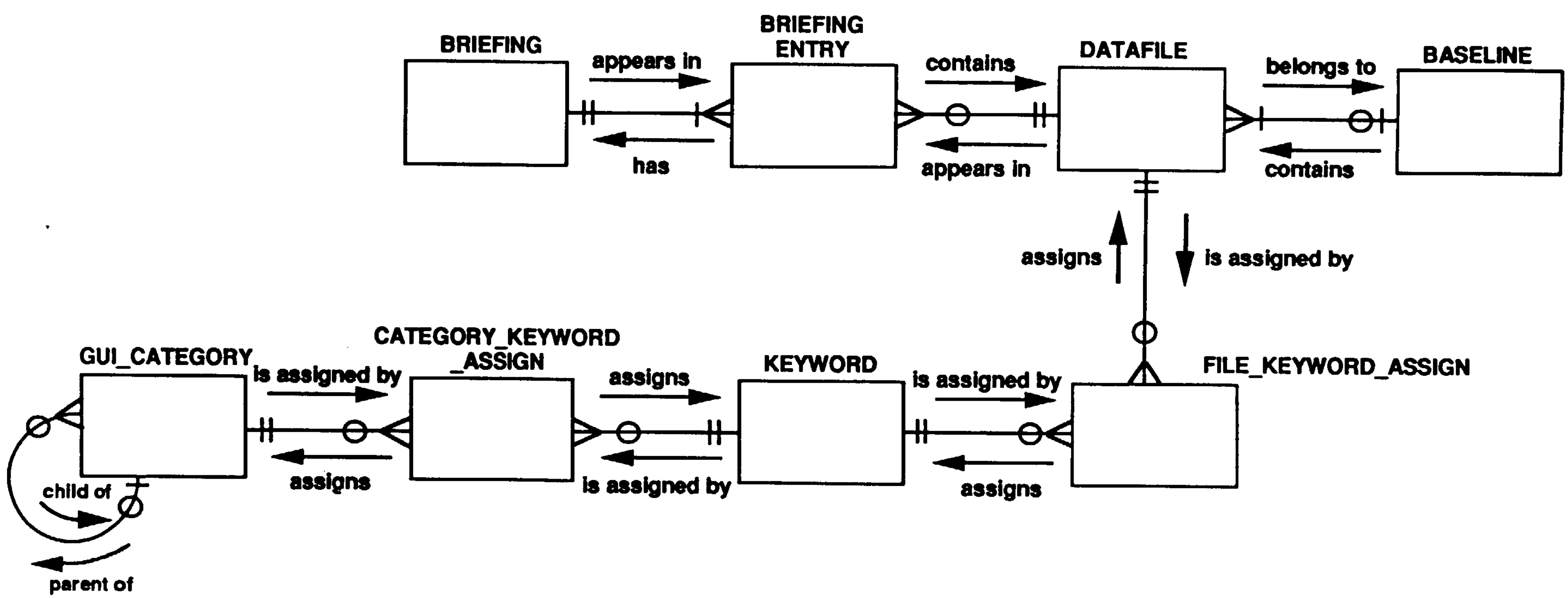
- Performed integration, testing, and demonstration. These activities included demonstrations of database use, database administration, and GUI access capability. Accomplished additional refinement of the system and added additional HLLV avionics requirements data to the database.

2.1.1 Database Development

As described in the previously delivered database model design document, the Propulsion System Database (PSDB) prototype, and the requirements document for a "fully-implemented" PSDB system delivered to NASA/MSFC under a previous contract, were used as the basis for deriving the functionality required for the "Electronic Filing System Index Database". The primary goal conceptualized by the PSDB prototype was that of providing NASA work groups with the means to easily store, review, retrieve, and reuse the volume of presentation material, much of it graphical, that is received for the variety of new projects under study at MSFC. Current methods result in this material being stored, usually on magnetic media such as diskettes, in a variety of locations with personal recollection as the only means to retrieve a particular graphic image or file. There has been no way to review the material other than loading each actual file, opening it, viewing, and closing. From this scenario, it can be seen that more efficient and reliable methods have been needed to effectively utilize this data. The PSDB requirements defined capabilities for storage of these graphical images and data files and also defined user interface capabilities for access to the data.

The database model design for the "Electronic Filing System Index Database" defined a relational database model, including entities, relationships, and attributes, as shown in Figure 2.1-1 and Table 2.1-1, that would fully implement the requirements specified in the Propulsion System Database Requirements Document. This relational model was intended to be a vendor independent database design that could then be used as an ideal model with which to compare and select a combined database/Graphical User Interface (GUI) tool or it could be used as a design with which to develop a custom database application using a relational database development tool (such as Oracle). This would also require an associated custom GUI with an interface package to the relational database.

Since the majority of the data files to be indexed in the Electronic Filing System database consist of graphic images used for presentation material, it was determined that an off-the-shelf commercial image database product consisting of an integrated database/GUI tool might be suitable as a basis for developing the electronic filing system application, if sufficient tailoring could be accomplished, and if it was functionally comparable to the relational model. An analysis of information available in trade publications and data solicited from vendors was performed. Evaluation versions of two tools were received and tested: Aldus' "Fetch" and Northpoint Software's "CompassPoint". The results of this study concluded that "CompassPoint" would sufficiently satisfy the requirements and could be tailored to include the necessary database attributes from the model. Thus, this tool was selected as the basis for developing the index database application.



ENTITY-RELATIONSHIP DIAGRAM
for
Electronic Filing System Database
(Normalized to 3NF)

(showing relationship name and degree)

Figure 2.1-1

Relational Data Base Entities and Attributes (primary keys are underlined)

- Entity Type: DATAFILE

  Attributes:

  <u>unique ID number</u>
  datafile name (not necessarily same as in pathname)
  version number (incremented for each newer file w/same name)
  full pathname (as the file is known to the computer operating sys.)
  subject
  creation date
  author
  formal document number (if applicable)
  date originally presented or delivered
  presenter who originally presented, or organization who delivered
  presentee who originally received
  owner currently responsible for file
  access - flag specifying pathname info. is restricted to owner
  application name (that created file, if applicable)
  baseline ID number (of baseline that the file belongs to,if none, file is in "working set")
  image (graphic "snapshot" for RDMSS that support this data type)

- Entity Type: BASELINE

  Attributes:

  <u>unique baseline number</u> (baseline "version")
  baseline data

- Entity Type: KEYWORD

  Attributes:

  <u>keyword ID number</u>
  keyword (name)

Table 2.1-1

- Entity Type: FILE_KEYWORD_ASSIGN

  Attributes: (Both attributes together form a composite primary key)

  <u>datafile ID number</u>
  <u>keyword ID number</u>

- Entity Type: BRIEFING

  Attributes:

  <u>briefing ID number</u>
  briefing name
  subject
  date of briefing
  person preparing/giving briefing
  primary person briefing for

- Entity Type: BRIEFING_ENTRY

  Attributes: (Both attributes together form a composite primary key)

  <u>briefing ID number</u>
  <u>datafile ID number</u>

- Entity Type: GUI-CATEGORY

  Attributes:

  <u>GUI category ID number</u>
  category name
  parent category number (null if top level)

- Entity Type: CATEGORY_KEYWORD_ASSIGN

  Attributes: (Both attributes together form a composite primary key)

  <u>GUI category ID number</u>
  <u>keyword ID number</u>

Table 2.1-1 (Continued)

2.1.2 Database Description

The CompassPoint tool combines an internal, optimized database management "back end" with a graphical user interface to form an integrated package for filing and tracking graphic images. The interface provides for a high degree of customization which allows the data base attributes from the model to be accommodated by the CompassPoint image database. Tables 2.1-2 through 2.1-5 describe the image database as implemented by SAT for the "Electronic Filing System Database" and delivered with HLLV avionics requirements study data. This definition represents only a suggested initial configuration of the database; it may be modified or updated as user experience dictates. Use of the database for storage and retrieval of information, as well as instructions for modifying setup, is thoroughly explained in the User's Manual, "CompassPoint for Windows Guide".

Image Base Name (up to 20 char.): NASA/MSFC EFS DB

Users:

First Name	Last Name	Dept.	Initial Password	User Level
System	Administrator		admin	10
	Engineer		engr	6
	Manager		mgr	2

Table 2.1-2 Database Name and Initial User Setup

Levels:

Security/User Changes	10
Setup Changes	10
Import Images	6
Change Records	10
Delete Records	10
Check Out/In	6
Copy/Export	6
Use Text View Reports	6
Use Text View	2
Use Multiview	1
Use Singleview	2
Use NAVITRACK view	2

Startup View: Multiview

Security ON

Table 2.1-3 Security Setup

Datafields:

Field	Label	Usage Notes
1	Name	descriptive name of image
2	Subject	
3	Category	
4	Author	creator of original
5	Doc. ID	
6	Presentation	purpose/date of original presentation
7	Presented by	
8	Presented to	
9	Pathname	
10	Orig. Format	
11	Appl. Created	application used to create original
12	Version	
13	Baseline	assign to all presented at same time
Date 1	Created	
Date 2	Presented	
Location of Physical	(fixed)	location of original (cabinet, etc.)

No Required Fields (all fields optional)

Table 2.1-4 Data Fields Setup

Edit/Add Options Setup:

Allow Batch Field Update	Yes
Allow Multiple Edit	Yes
Allow Auto Import	Yes
Store Full Res. File	Decide on import
PhotoCD Resolution	Decide on import

NAVITRACK/Check Out Setup:

Allow Editing of NAVITRACK records	Yes
Allow Deleting of NAVITRACK records during Edit	Yes
Allow Deleting of NAVITRACK records on Record Delete	Yes
Create NAVITRACK records on Copy/Export	Yes
Create NAVITRACK record on Check Out/In	Yes
Use check out/in procedures	Yes

Text View Report Setup:

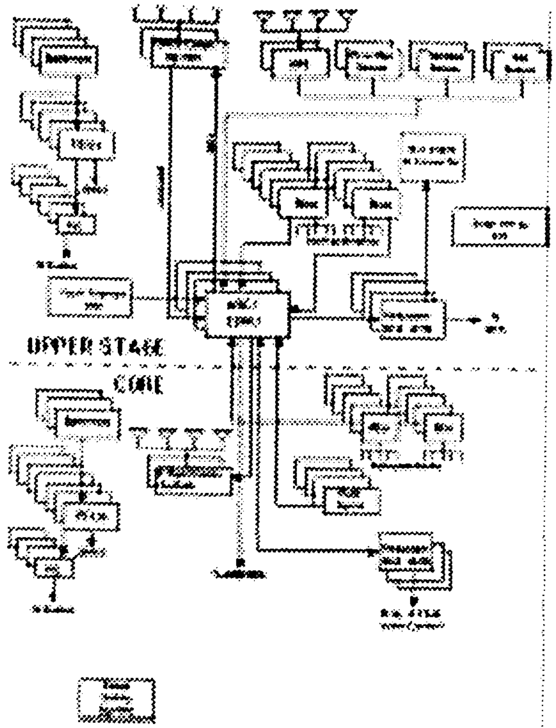
Use Defaults

Table 2.1-5 Edit, NAVITRACK, and Text View Setup

2.1.3 Conclusions

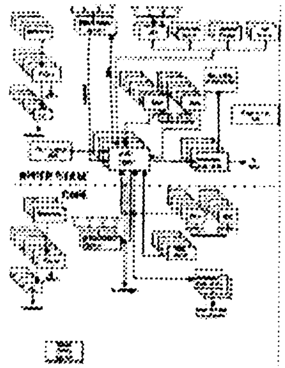
As a result of this task, SAT believes that the use of a graphic database for providing indexing information for delivered study data is a viable method for achieving storage and access efficiency. With the CompassPoint tool's ability to utilize multiple image bases, a contractor using the tool could create a complete index data base, containing the image "thumbnails", for delivery to a project team (along with the full-resolution original images). Team members could then log on to the image base directly, without having to import the images into another data base. An alternative would be for the project team to accept an electronic copy of all the graphic images, then import them into the project data base using the CompassPoint "Auto Import" feature to automatically import multiple images. Images could also be added to the database individually. The time required to import images and data into the attribute fields for each image would be minimal compared to the time saved in searching, reviewing, and retrieving the images. The CompassPoint tool's extensive searching and sorting capability can be used to narrow the number of images for viewing or to find a specific image and then arrange the image subset in a useful order. After locating an image subset, images can be viewed or printed in single or multiple mode. An example printout for single mode is shown in Figure 2.1-2 and for multiple mode is shown in Figure 2.1-3. With the networking and file server capability, the added value of having an organized, manageable database containing a project's graphic data available to all members of a distributed project team would be well worth the cost of the tool.

In summary, SAT believes the improved use of available resources provided by this approach would be a significant benefit to a project team's data management requirements.

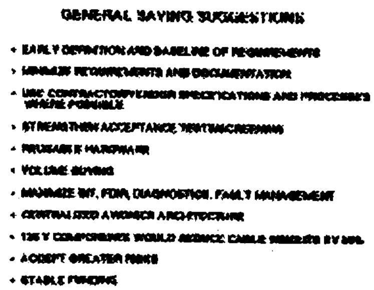


Name: Block Diagram
Subject: Architecture
Category: HLLV
Author:
Doc. ID: 01
Presentation: Mgt. review-3/31/93
Presented by: Mary Harris
Presented to: Sam Haley
Pathname: SAT:avionics:BLOCK DIAGRAM
Orig. Format: MacDraw drawing
Appl. Created: MacDraw II 1.1
Created: 02/27/1993
Presented: 02/27/1993
Version: 1
Baseline: 1

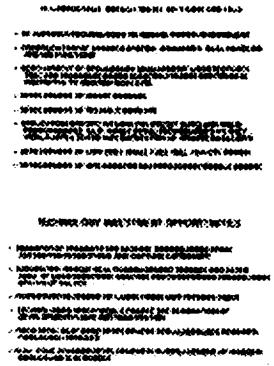
Figure 2.1-2



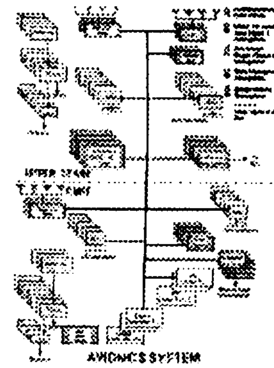
Block Diagram



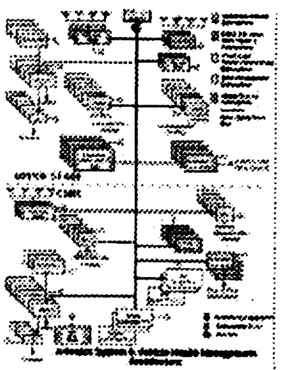
Cost Savings Suggestions



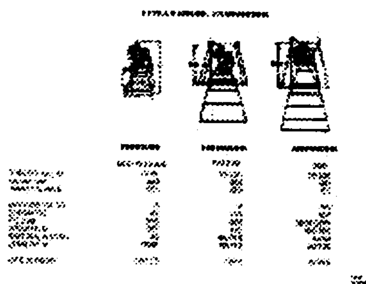
Investment Opportunities

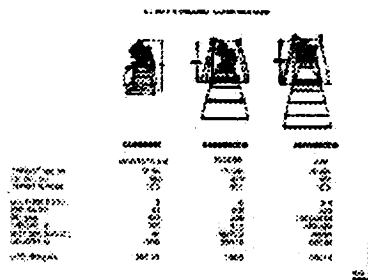


Block Dia.-New tech.



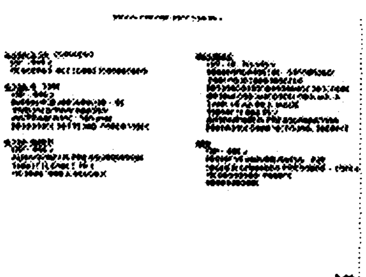
Block Diagram-Update



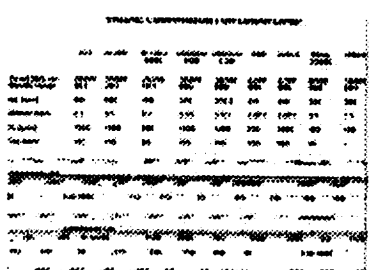


Engine Comparison

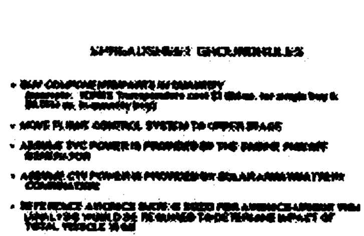
Health Management



Engine Descriptions



Lander Engine Comparison



Spreadsheet Groundrules

Figure 2.1-3

2.2 HLLV AVIONICS REQUIREMENTS TASK

The HLLV avionics requirements task consisted of a study effort to investigate avionics system requirements and architectural concepts for a heavy lift launch vehicle. SAT's significant accomplishments for this task include the following:

- Completed an initial start-up phase. During this phase, held meetings with appropriate organizations and personnel to establish ground rules and direction for the study and to define any limiting assumptions. Agreed that the focus of the study would be incorporation of Vehicle Health Management (VHM).

- Began a data gathering process. Analyzed documentation for relevant information to derive avionics requirements related to VHM. Held meetings with MSFC personnel in the effort to drive out the requirements data and incorporate differing viewpoints.

- After the accumulation of information was begun, started a data analysis activity. This activity involved the evaluation of data received and the refinement of requirements based on that data. Compared HLLV avionics systems requirements to NLS, including VHM.

- Prepared an interim report and submitted it to the NASA/MSFC HLLV Avionics Group.

- Continued the iterative processes of data gathering and data analysis for the remainder of the study.

- Developed the study information and analysis into a conceptual architecture including requirements of HLLV avionics systems incorporating VHM.

2.2.1 Requirements

The primary focus of this study was to concentrate on the implementation effects of Vehicle Health Management (VHM) on the avionics system of a Heavy Lift Launch Vehicle (HLLV). It should be noted, however, that the concepts presented in this study are applicable to the total family of next generation launch vehicles whether they are expendable, reusable, single-stage-to-orbit or multi-stage. The primary driver for VHM is to enhance the probability of mission success and at the same time reduce operations cost throughout the full operations life cycle of the launch vehicle. In this context, VHM should be considered as the "onboard" subset of an overall Health Management System (HMS). The HMS will provide the integrated ground and vehicle function of managing the full launch complex system process. The function of VHM is to determine vehicle health and maintain the vehicle in a functional and safe condition through the full mission life cycle.

This study is based on the work accomplished by the NLS Avionics product development team as documented in **National Launch System Avionics Architecture System Definition Document** dated January 31, 1992. From this, a generic conceptual architecture was developed and is depicted in Figure 2.2-1. In this concept, the guidance/navigation/control and command & data handling functions are integrated into the avionics system computer (GN&C/ASC/CDHU). This integrated function is physically located in the upper stage and provides control for the entire vehicle through redundant fiber optic data busses which provide the command and data path to the ancillary avionics components. The quad redundant architecture operates as a triple redundant set with the fourth "string" as a hot spare. This arrangement allows the avionics system to be tailored to be triple redundant by simply not installing the fourth set of components if mission requirements do not require the reliability and dependability of quad redundancy. This may be the case for a mission that only requires insertion into low earth orbit. For a longer duration mission such as that required by a single-stage-to-orbit or a reusable launch vehicle, the full quad redundant implementation may be necessary.

Avionics systems reliability/dependability is a function of operating time. Longer operating times widen the window of opportunity for a failure to occur. The concept of redundancy is then essential in order to provide dependable mission performance. Triple redundancy is much simpler to implement since a two-of-three vote is all that is required to detect a failure. For this reason, a triple redundant system was chosen as the minimum implementation for an avionics system. Additional redundancy, such as the addition of a fourth string does not bring additional mission assurance except in the case where the fourth string is used as a hot spare. Figure 2.2-2, which is from a Rockwell study concerning STS preflight failures, presents this argument in graphical form. Although this addresses preflight reliability, it can be extrapolated to inflight reliability as shown in the **Space Avionics Architecture Definition Study** done for the NASA Strategic Avionics Technology Working Group (SATWG) by Boeing (NAS1-18762). With a properly defined health management implementation, the individual components of the spare string can be used to replace only its failed counterpart and not require a full string to be switched out when a component fails. From a preflight standpoint, an arrangement such as this can provide the capability to launch with a failure (no mission scrub) and still meet mission success criteria for an ascent to orbit mission.

To restate, the primary reason for VHM is to reduce total mission operations cost. Inherent in this statement is also the enhancement of the probability of mission success. In order to accomplish these objectives, VHM must be able to not only determine the health of the vehicle but also manage the state of the vehicle such that it is in compliance with mission specifications. Thus, during ground operations VHM will provide an assessment of current vehicle health and provide a prognosis of future health. VHM will also provide fault detection and isolation to a Line Replaceable Unit (LRU). During prelaunch and terminal countdown VHM will ascertain the vehicle's readiness and initiate automatic safeing if a no-go situation is diagnosed or prognosed. This is accomplished in real-time and is especially significant during engine start and thrust build-up for liftoff. During the flight portion of the mission whether it be ascent, orbital, descent, etc. VHM provides for maintenance of vehicle health by accomplishing fault detection, isolation and recovery. It supports such actions as engine out or engine restart decisions as well as emergency detection for manned flights and collision avoidance functions for docking maneuvers.

2.2.2 Conceptual Architecture

Figure 2.2-3 shows a conceptual VHM architecture superimposed on the avionics architecture that was previously described. In this concept, VHM is a function that is embedded in the avionics system. VHM is hierarchical and distributed in the sense that at the lowest tier each LRU has health determining capability. This may be in the form of built-in-test or other means of verification. At the next level, each subsystem has the capability to determine and manage its own health. The subsystem health manager function is able to take autonomous actions to disqualify failed LRU's within the subsystem, to reconfigure its complement of LRU's and to bring the spare units on line as necessary to maintain the subsystem at the necessary level of functionality. Subsystem health status and all remedial actions taken are reported to the next tier, the vehicle health manager. The vehicle health manager perform a supervisory function in statusing and controlling the overall health of the vehicle. Vehicle health manager function may override the actions of lower tier subsystems and is ultimately responsible for the overall functionality and safety of the vehicle.

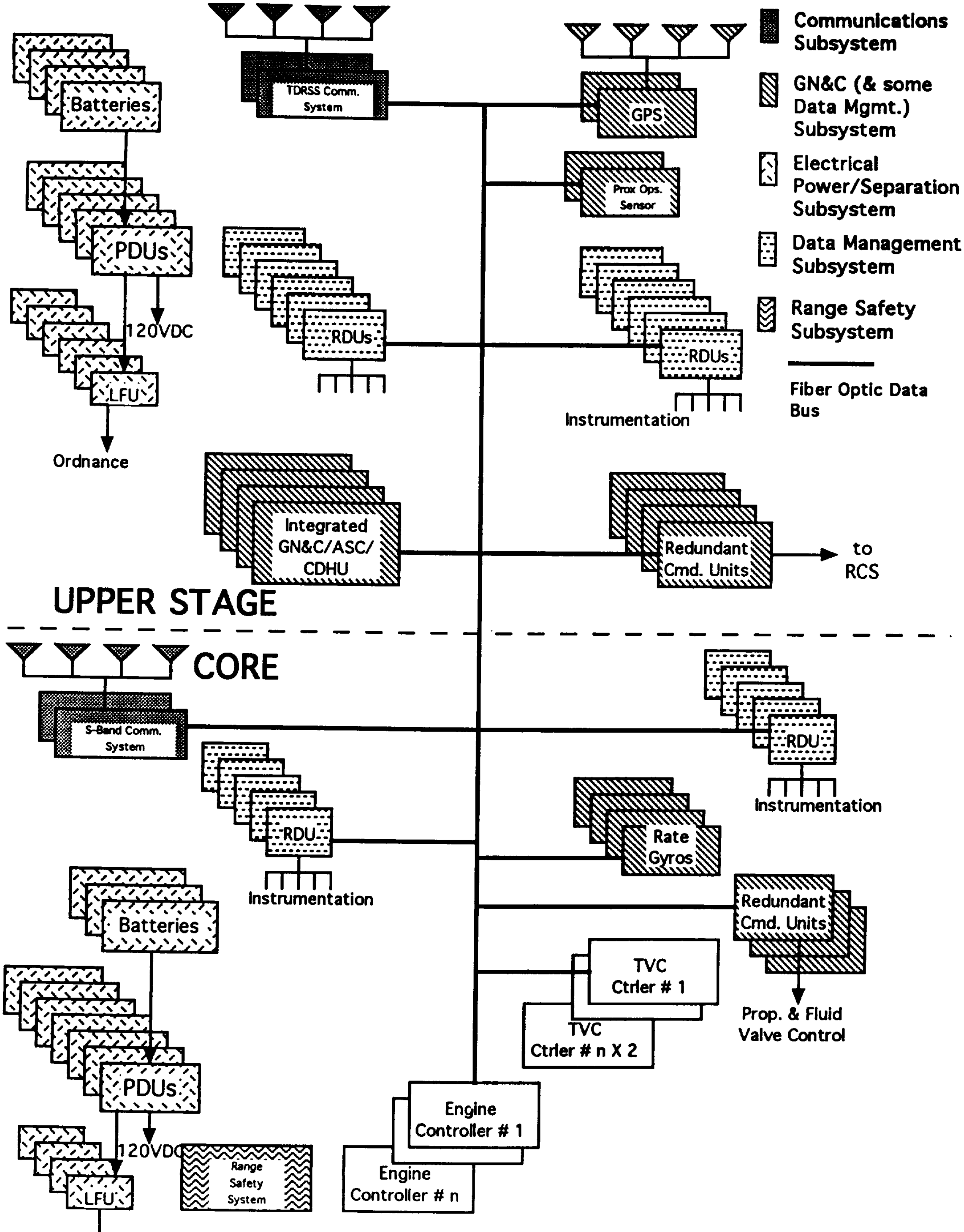
In order to accomplish VHM as described, the implementing avionics system as well as the overall vehicle must be originally designed for health management. This implementation assumes an "intelligent" approach to vehicle health management based on knowledge of component and system performance characteristics. In past vehicle designs, the practical state of the art for the implementing flight hardware and software did not support this type approach. However, significant technological advances in avionics system components (hardware and software) have been made since the last major NASA vehicle design. The current state of the art does now support appropriate flight systems implementation.

Knowledge based software and its demand on flight system computational resources and the need to perform in real time is the major obstacle to implementing vehicle health management as described. Recent advances in a polynomial network approach to knowledge based systems have been made by **AbTech Corporation**, 508 Dale Avenue, Charlottesville, Va. with their development of a commercially available tool set for abductive polynomial network synthesis. This tool set includes both an abductive information modeler as well as a generator for the executable. The Abductive Information Modeler (AIM) provides a practical supervised empirical learning tool for synthesizing abductive polynomial networks from data bases of input and output values for example situations which may be analytically or empirically derived. Supplied with sample data from diagnostic or prognostic evaluations, AIM synthesizes polynomial networks and encodes them into C source code that can be readily incorporated into application software. The network size, connectivity and parameter values are all determined automatically by AIM. The network models emerging from the AIM synthesis process are robust and compact transformations implemented as layered networks of feed-forward functional nodes. With the hierarchial and distributed approach described for the vehicle health monitoring system, the network size and resulting C code can be kept within bounds of low cost flight processors. The currently available 286 class microprocessor is capable of performing these functions. development and evaluation process in the MAST.

As previously stated, The overall health management function includes not only the vehicle portion but also a ground systems implementation. A knowledge based approach for Ground Health Management that is complementary to the vehicle health management implementation is recommended. There are currently available commercial products which are capable of providing the necessary functionality. Two such products are used currently at several NASA Centers for various applications in mission management. These are, **G2** developed by the Gensym Corporation, 125 Cambridge Park Drive, Cambridge, MA. and **RT works**, developed by Talarian Corporation, 444 Castro Street, Suit 140, Mountain View, CA.

2.2.3 Conclusions

In order to field a Vehicle Heath Management system as described, considerable research and development work is necessary which includes testing of candidate hardware/software implementations in a simulation laboratory environment. A laboratory with the capacity to accomplish this activity currently exists at Marshall Space Flight Center. The Marshall Avionics System Testbed (MAST) is a world class simulation facility developed to demonstrate, test and evaluate advanced avionics systems and components. One of the expressed objectives of the MAST is to provide a testbed to develop and confirm the viability of Vehicle Health Management implementations. With the current availability of the implementing avionics systems technology and the NASA activity towards the next generation of launch vehicle, the time is right to begin the development and evaluation process in the MAST.



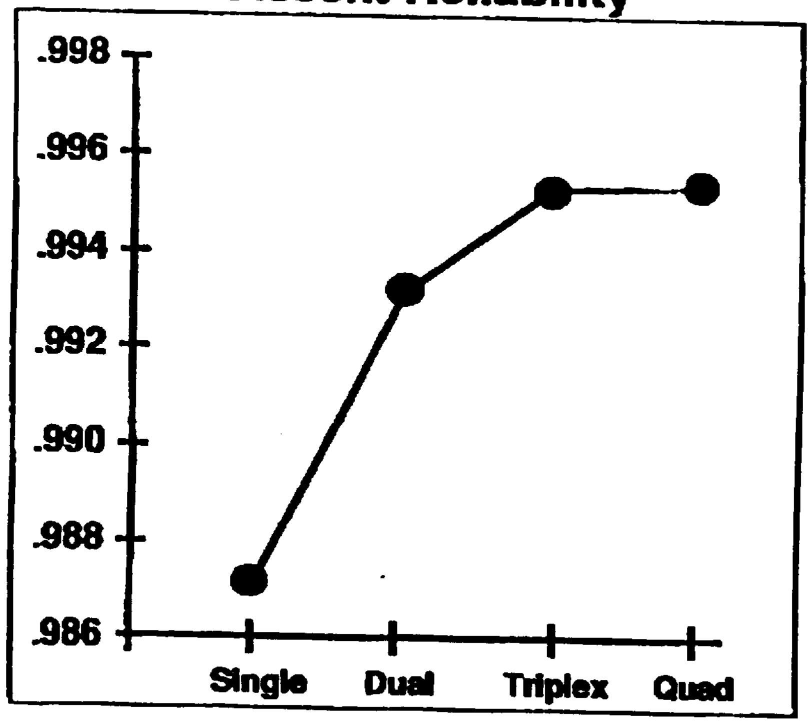
AVIONICS SYSTEM

Figure 2.2-1

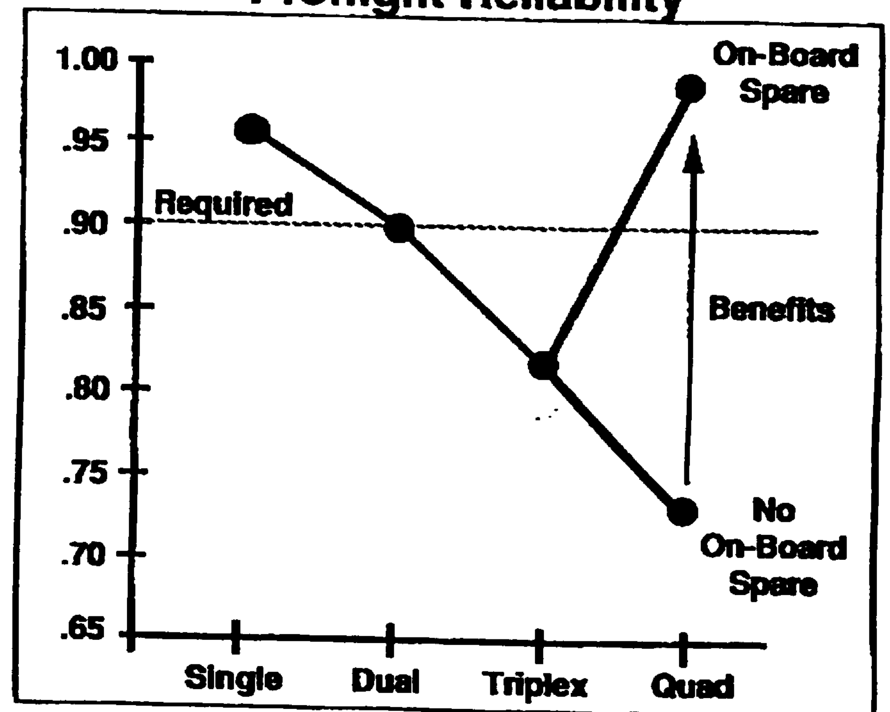
Balancing Reliability and Dependability

Reliability Model Data

Ascent Reliability



Preflight Reliability*



* Mechanical systems not modeled as active during pre-flight phases

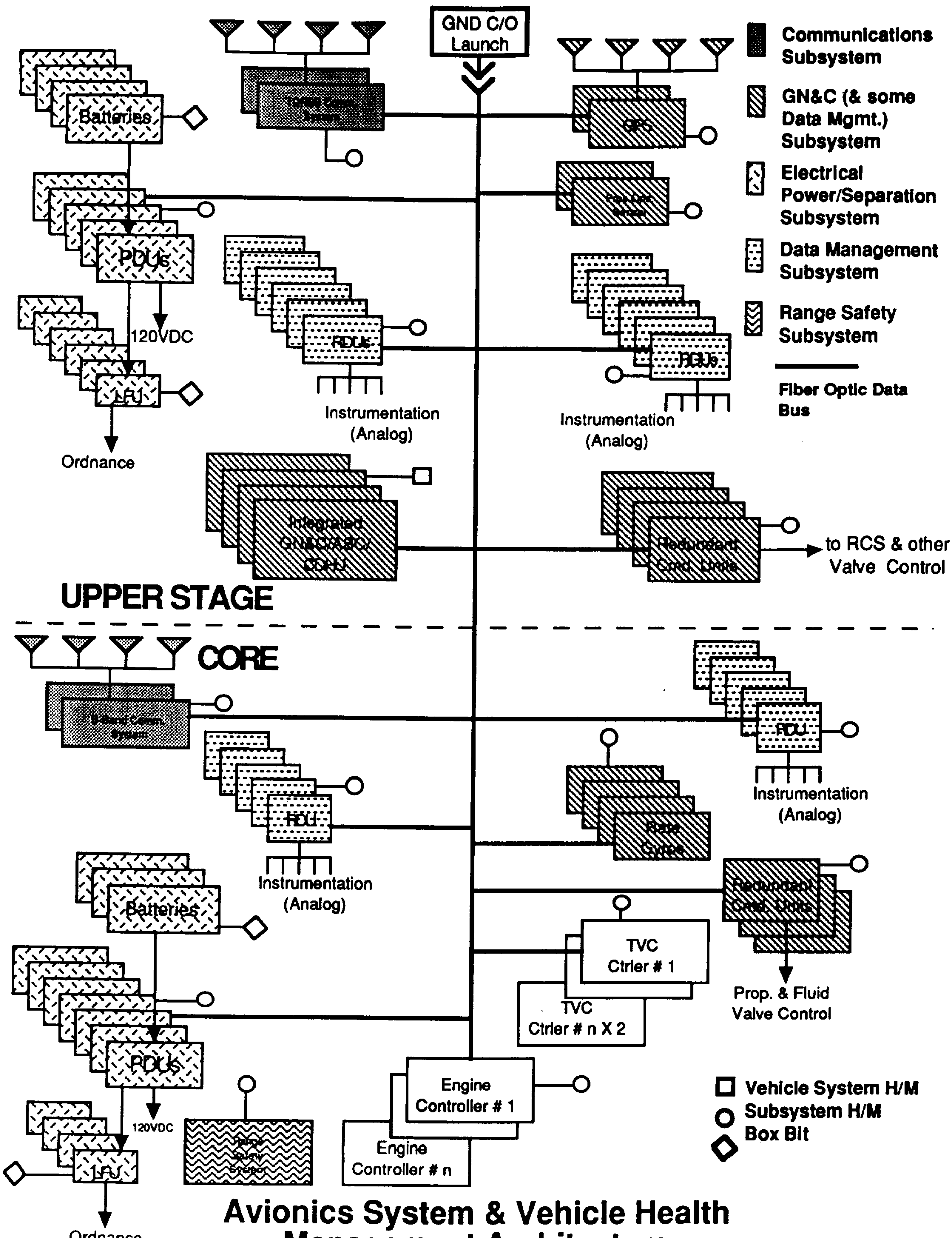
Empirical Data*

STS pre-flight failure data:

- **4 preflight failures due to avionics through STS-39 (3 required remove & replace action)**

* RI study on STS launch delays, 7/91

Figure 2.2-2



Avionics System & Vehicle Health Management Architecture

Figure 2.2-3